ESD-TDR-63-206

ESD-TDR-63-206
(MITRE TM 3308)

RESOURCE ALLOCATION IN A PERT NETWORK UNDER CONTINOUS ACTIVITY TIME-COST FUNCTIONS

TECHNICAL DOCUMENTARY REPORT NO. ESD-TDR-63-206

AUGUST 1962

E. B. BERMAN

ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

(Prepared under Contract No. AF 33(600)-39852 by The MITRE Corporation, Bedford, Massachusetts)

Best
Available
Copy

ESD-TDR-63-206

Technical Memorandum TM-3308

Sheet 1 of 10 Sheets

Contract No. AF33(600)-39852

SUBJECT: RESOURCE ALLOCATION IN A PERT NETWORK UNDER CONTINUOUS ACTIVITY TIME-COST FUNCTIONS

TO: N. Waks

FROM: E. B. Berman

DEPT.: D-18

DATE: 15 August 1962

APPROVED: N. Waks

N. Waks

THE MITRE CORPORATION

Bedford, Massachusetts

ISSUED AT:

ABSTRACT

This memorandum describes a conceptual model which allocates resources in a PERT network, the activities of which are subject to continuous concave-upward time-cost functions, in such a way as to achieve a minimum cost solution for a given completion date for the program described by the network. A parametric solution for the minimum cost solution at various total program times is also briefly described. The effect of uncertainty in the time of completion of an activity has also been investigated in a highly simplified model; and a tentative conclusion is presented that such uncertainty tends to divert resources toward the earlier part of the program, as compared to the solution under certainty

TABLE OF CONTENTS

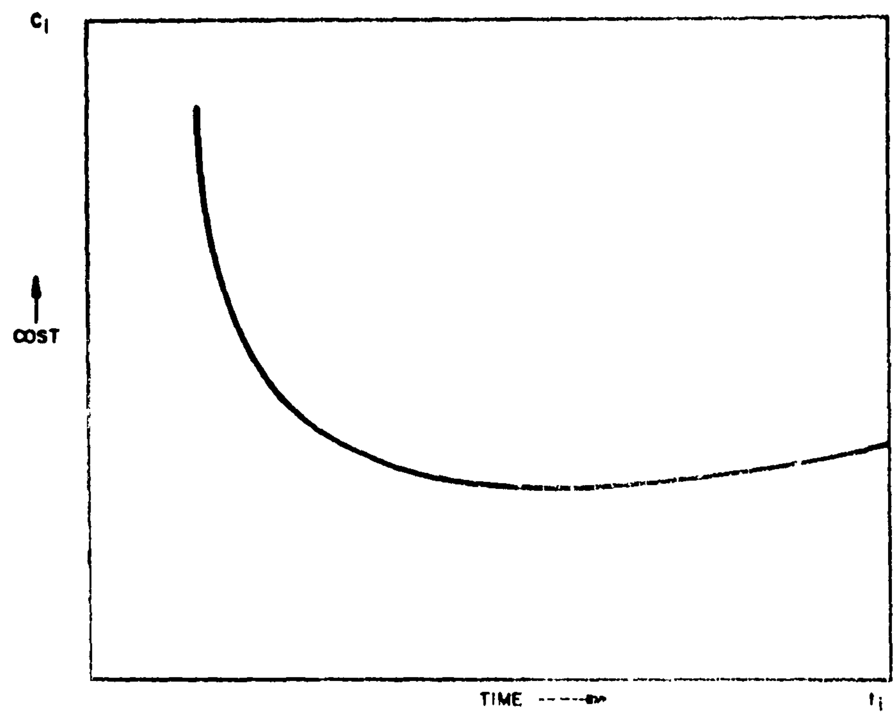
1.0 INTRODUCTION

This memorandum describes a conceptual model for allocating resources, represented by dollar cost, in a PERT network to achieve a given completion date for the program at minimum cost.

It is assumed that each activity or task in the network is subject to a continuous upward-concave time-cost relationship as illustrated in Figure 1. Note in Figure 1 that the cost approaches infinity in any real

Figure 1

IA-9515



UPWARD — CONCAVE TIME-COST FUNCTION FOR A TYPICAL ACTIVITY

activity (dummy activities are excluded) as time approaches a minimum feasible time. There is, however, a minimum to the function representing the fact that costs will turn up beyond a certain point, since further increases in time imply increasingly inefficient use of facilities and equipment and increasing liability to engineering change, which cannot be compensated for by increasing efficiency in the use of manpower. Further increases in time must eventually imply even decreasing efficiency in the use of manpower. It is assumed that slack is permissible in the network, so that in fact it is never necessary to proceed to the right of the minimum.

It is assumed that there is a given time T for the completion of the total program represented by the network, and that this total time is feasible. The memorandum is addressed to the problem of balancing the allocation of time within the network (and hence the allocation of resources) to minimize the cost of achieving the program in time T.*

In this paper, a decision system is set forth in which the time for the completion of an activity is determinable, although variable as a management decision. Note is also taken of the direction of bias which results from failure to consider the random element which actually exists in activity time.

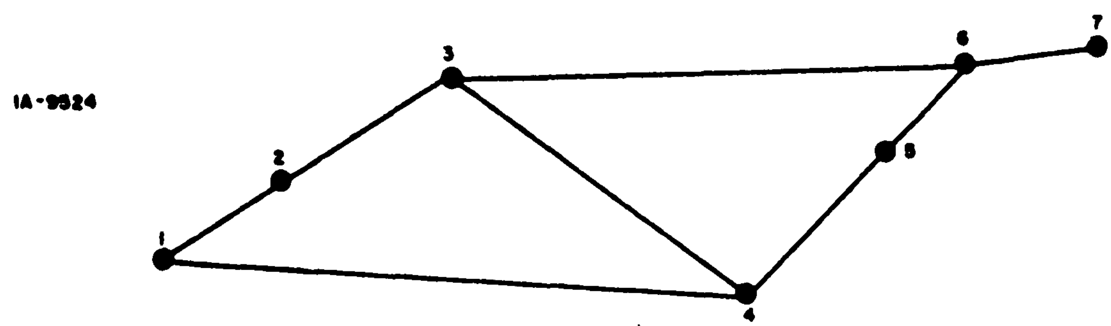
The approach under certainty is similar to those described in the literature [1] [2] and [3] except that continuous functions are used here in place of the segmented linear time-cost functions and that a new iterative algorithm, more suitable for continuous functions, is introduced. The continuous functions permit a more general solution of the resource allocation problem and offer the greater sensitivity necessary to study the effects on the decision arising from uncertainty in the time of completion of an activity.

2.0 BALANCING A NETWORK UNDER CERTAINTY

2.1 Balancing Time Along A Serial Path

A serial path is defined as a path of two or more activities which must be performed in sequence and which are not related in sequence to any activities not on the path except at the start of the first activity and at the end of the last. In Figure 2, points 1, 2, and 3 and points 4, 5, and 6 represent two serial paths.

Figure 2

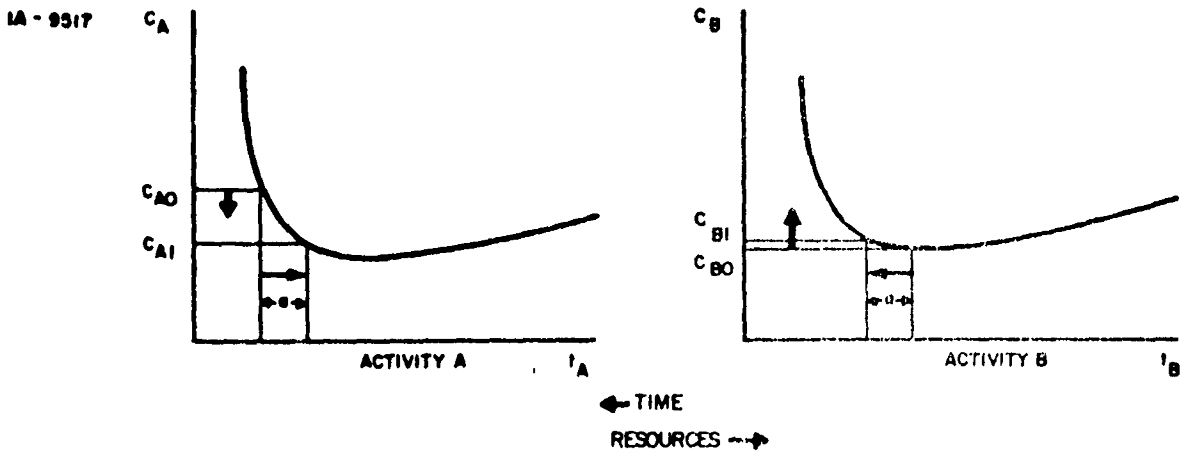


NETWORK WITH SERIAL PATHS

*No consideration has been given here to manpower loading, a serious and difficult problem for the real-world decision-maker. The failure to consider this problem undoubtedly limits the usefulness of this model.

The balancing of time, or the resource allocation, is first considered along a serial path, in which the path as a whole must be completed in time T_i. First, a solution in which all activities along the serial path can be located at or to the right of the minima of the time-cost functions is not considered; for in this case, since slack is permitted in the network, the minimum cost solution is such that each activity along the serial path is exactly at the minimum of its time-cost function, and no time-cost tradeoffs are brought into issue. The more interesting case is the one in which all activities along the serial path are to the left of their time-cost minima. In this case, the cost of the serial path itself can be minimized by reallocating resources in such a way that the time-cost functions all have equal slopes along the path (they are all negative, of course) and such that the total time of all activities on the path is equal to T_i. A reallocation towards equality of slopes will always lower the total cost of the program, as illustrated in Figure 3.

Figure 3



THE ALLOCATION OF TIME FROM ACTIVITY B TO ACTIVITY A

Here, an allocation of time from activity B with its shallow time-cost slope, towards activity A with its steep time-cost slope, retaining a constant sum of time in the two activities, will necessarily reduce the total cost of the two activities without changing program time T_i. This follows from the fact that the saving from increasing time in activity A will be greater, by reason of its steeper slope, than the cost of decreasing the time in activity B. In Figure 3, the transfer of a unit of time from activity B to activity A raises the cost in B only from C_{BO} to C_{B1}, but lowers the cost in A from C_{AO} to C_{A1}. It should be noted that the transfer of resources is in the opposite direction from the transfer of time, i.e., from activity A to activity B.

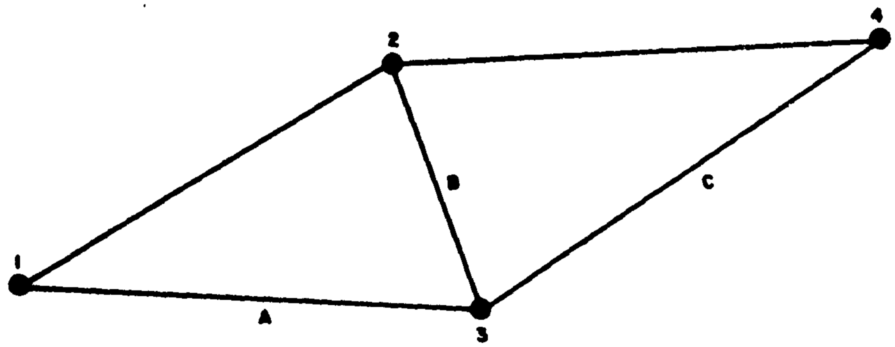
From this it is possible to conclude that, along a serial path, the time-cost slopes of all activities should be equal to each other and less than or equal to zero.*

2.2 Balancing Time at a Complex Junction

The optimum location of a complex junction in time is now considered, with the other junctions in the network held constant in time. A complex junction is defined as one at which more than one activity originates and/or terminates. The network illustrated in Figure 4 contains two complex junctions, 2 and 3.

Figure 4

1A-9823



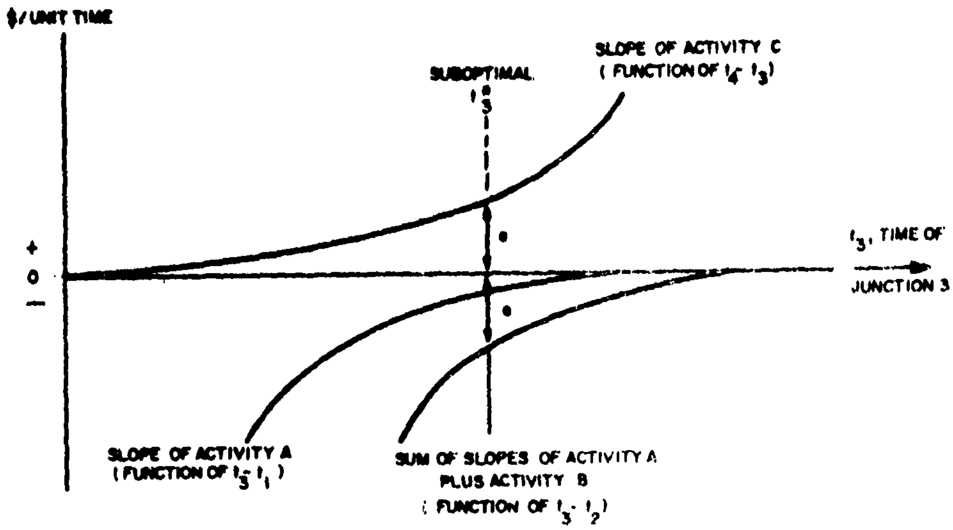
A COMPLEX JUNCTION

Figure 4 illustrates the optimum location of junction 3 in time, t_3, with junctions 1, 2, and 4 held fixed in time. Here, as t_3 is moved closer to t_4, the time in two activities, A and B, is increased; and the time in one, activity C, is decreased. Thus the point of balance or suboptimum location of junction 3 in time (t_3) is such that the incremental savings in activities A and B would exactly equal the incremental cost in activity C as a function of the movement of t_3 toward the end of the program. This situation is illustrated in Figure 5 on the following page; any movement of t_3 to the right from t_3^*, the point of balance, would raise the cost in activity C more than it would lower the cost in activity A and activity B, since in such a case the slope in activity C is greater than the sum of slopes in activities A and B in absolute value. Conversely, any movement left from t_3^* would raise the cost in activities A and B more than it would lower the cost in activity C, since in this case the sum of slopes in A and B exceeds the slope of activity C, in absolute value.

*Andrew Vazsony[4] comes to the conclusion that time-cost slopes should be equal in subactivities of a single activity.

IA - 9518

Figure 5



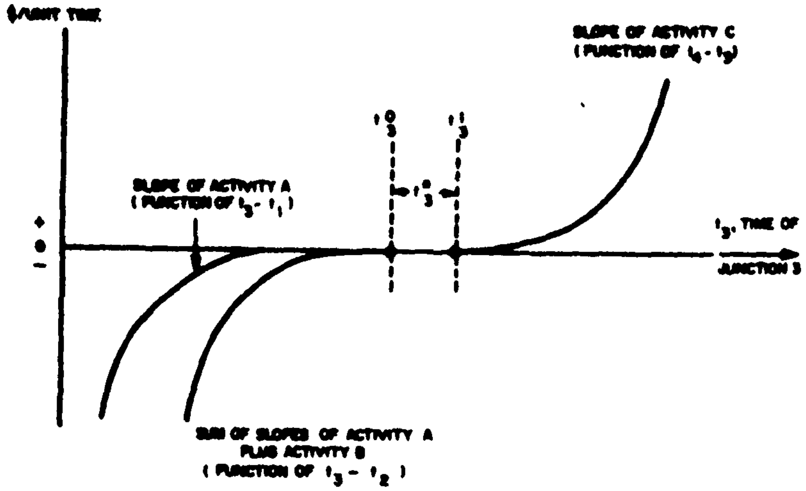
SLOPES OF TIME-COST FUNCTIONS OF A COMPLEX JUNCTION, SHOWING t_3^*, POINT OF BALANCE

It should be noted that although the slopes here are shown concave upward, in absolute value, the existence of a solution depends on the monotonicity of the slopes, not on their concavity; and monotonicity can be implied from the assumption that the time-cost functions themselves (Figures 1 and 3) are concave upward.

With a slack junction, in which it is possible for all activities to reach their time-cost minima, the point of balance of the junction is indeterminate, although the minimum cost is determinate, and is equal to the sum of time-cost minima of the relevant activities. This situation is illustrated in Figure 6 on the following page.

In Figure 6, the sum of slopes of the time-cost functions for activities A and B reach zero at t_3^0 and the slope of the time-cost function for activity C reaches zero at t_3^1. Thus from t_3^0 to t_3^1 the sum of the cost of activities A, B and C remains constant at the sum of the time-cost minima and the sum of the slopes of A and B is equal to the slope of activity C at zero.

Figure 6



SLOPES OF TIME-COST FUNCTIONS IN A COMPLEX JUNCTION WITH SLACK, SHOWING RANGE OF SUBOPTIMAL SOLUTION t_3^*

It is possible to conclude from the above analysis that the suboptimal location in time of one junction in a network, with the other junctions held fixed in time, is such that the sum of time-cost slopes of activities leading into the junction will be equal in value and opposite in sign to the sum of time-cost slopes of activities leading out of the junction, or that suboptimal t_j, with t_is and t_ks held fixed, is such that:

$$\sum_i \frac{dC_{ij}}{dt_j} + \sum_k \frac{dC_{jk}}{dt_j} = 0 \qquad (1)$$

where C_{mn} $(t_n - t_m)$ is the time-cost function of the activity commencing at m and terminating at n. Equation (1) is the partial derivative of the cost function of the total program with respect to t_j; there is one such partial derivative for each internal junction.

2.3 Balancing an Entire Network

2.3.1 Methodology

By extension from the discussion of section 2.2, it is possible to define a balanced network, providing the minimum cost solution for a total program time T, as one in which each internal junction satisfies equation (1) and in which $T = t_n - t_1$ for a network of n junctions. Although it is possible that an analytical simultaneous solution might be obtainable for an entire network by setting the partial derivative for each internal junction (equation 1) to zero and solving simultaneously for each internal junction, it is suggested that the network be solved iteratively, moving either forward or backward through the network, balancing a single junction at a time with each other junction held fixed at its last computed suboptimum. Such an algorithm would clearly converge to a solution, since there would be a tendency in each junction to underadjust because each neighboring junction would be held fixed.

The following specific iterative algorithm is suggested;

1. Set each activity at a fairly steep slope of the time-cost function, leaving such slack as is required at the latter part of the network. Introduce additional slack as required. This step provides an initial feasible solution.

2. Balance junctions in succession, moving backward through the network.

3. Iterate step 2 until an acceptable degree of balance is obtained at each junction.

This process would tend to provide a steady progression of the t_js for intermediate junctions toward the end of the program until convergence is obtained.

2.3.2 A Hypothetical Example

For this example, time-cost functions have been specified in the general form:

$$C_{ij}(t_{ij}) = a_{ij} + b_{ij}\, t_{ij} + \frac{c_{ij}}{t_{ij}-d_{ij}}; \quad d_{ij} < t_{ij} \leq \sqrt{\frac{c_{ij}}{b_{ij}}} + d_{ij}$$

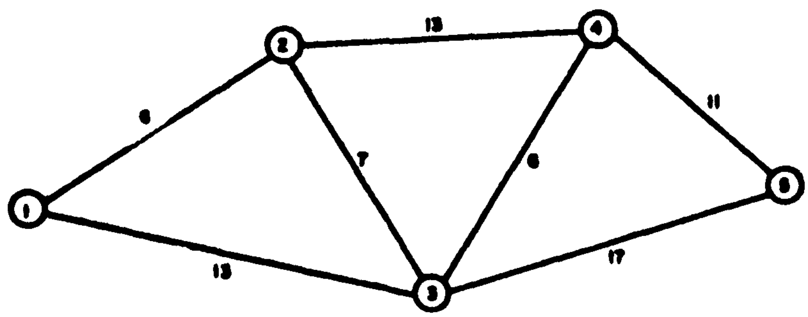
$$C_{ij}(t_{ij}) = C_{ij}\left(\sqrt{\frac{c_{ij}}{b_{ij}}} + d_{ij}\right) \text{ for } t_{ij} > \sqrt{\frac{c_{ij}}{b_{ij}}} + d_{ij} \tag{2}$$

where a_{ij}, b_{ij}, c_{ij} and d_{ij} are constants and t_{ij} is the time duration of activity ij. (It should be noted, incidentally that whereas t_j, the time of junction j, is a point in time, t_{ij}, the time of activity ij, is a span of time). The term a_{ij} represents a fixed cost component; the term $b_{ij}\, t_{ij}$ represents a linear cost component, proportional to t_{ij}, the time of the activity; and the term $\frac{c_{ij}}{t_{ij}-d_{ij}}$ represents the portion of the cost which is inversely related to the time of the activity, reaching infinity as $t_{ij} \to d_{ij}$ from above and approaching zero as $t_{ij} \to \infty$. For $t_{ij} > \sqrt{\frac{c_{ij}}{b_{ij}}} + d_{ij}$, the minimum of the C_{ij} (t_{ij}) function, the cost remains at the minimum of the function, representing the fact that slack is permissible in the system. The second term roughly represents the impact on capital cost and the third term roughly represents manpower cost. The first term may be associated with the cost of materials.*

The network represented in Figure 7 was used in the calculation, together with the constants presented in Table I.

Figure 7

IA-9882



PERT NETWORK USED FOR HYPOTHETICAL EXAMPLE, SHOWING THE t_{ij}'s FOR AN INITIAL FEASIBLE SOLUTION

Table I

CONSTANTS

Activity	a_{ij} ($'S)	b_{ij} ($/DAY)	c_{ij} ($-DAYS)	d_{ij}(DAYS)	Initial t_{ij}(DAYS)
1-2	1,000	100	5,000	5	6
2-3	1,000	80	4,000	6	7
1-3	1,000	50	3,000	10	13
2-4	1,000	150	8,000	8	13
3-4	1,000	110	6,000	5	6
4-5	1,000	120	7,000	7	11
3-5	1,000	100	6,000	7	17

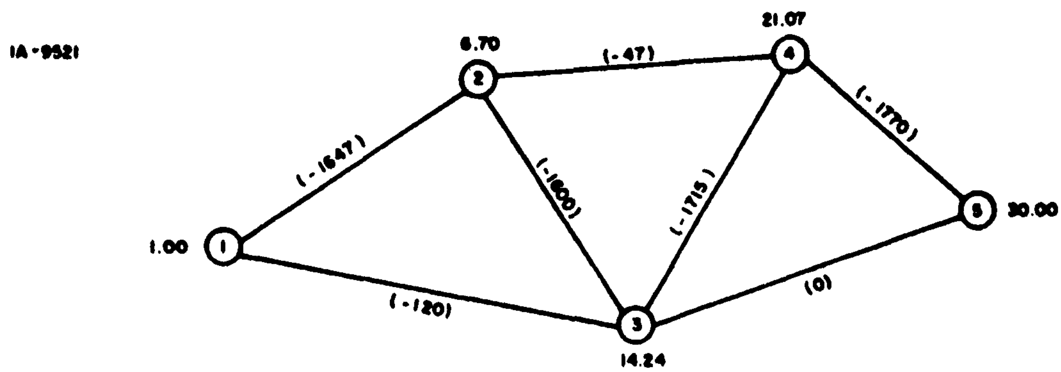
T, total program time = 30 days

Constants Used in Hypothetical Example

*These time-cost functions are similar to those used in reference 5 except that the time-cost functions used there omit the linear component.

The initial feasible solution is shown in Figure 7. The network was then balanced,* and activity times and time-cost slopes were derived. These are presented in Figure 8 and Table II. The time-cost slopes are shown in parentheses in Figure 8.

Figure 8



THE HYPOTHETICAL NETWORK AFTER BALANCING

Table II

Activity	Optimal t_{ij} (Days)	Time-Cost Slopes At Balance ($/Day)
1-2	6.70	-1647
2-3	7.54	-1600
1-3	14.24	- 120
2-4	14.37	- 47
3-4	6.83	-1715
4-5	8.93	-1770
3-5	15.76	0

Optimal Activity Times & Time-Cost Slopes For Hypothetical Network

*The calculations were performed on a PACE Analog Computer.

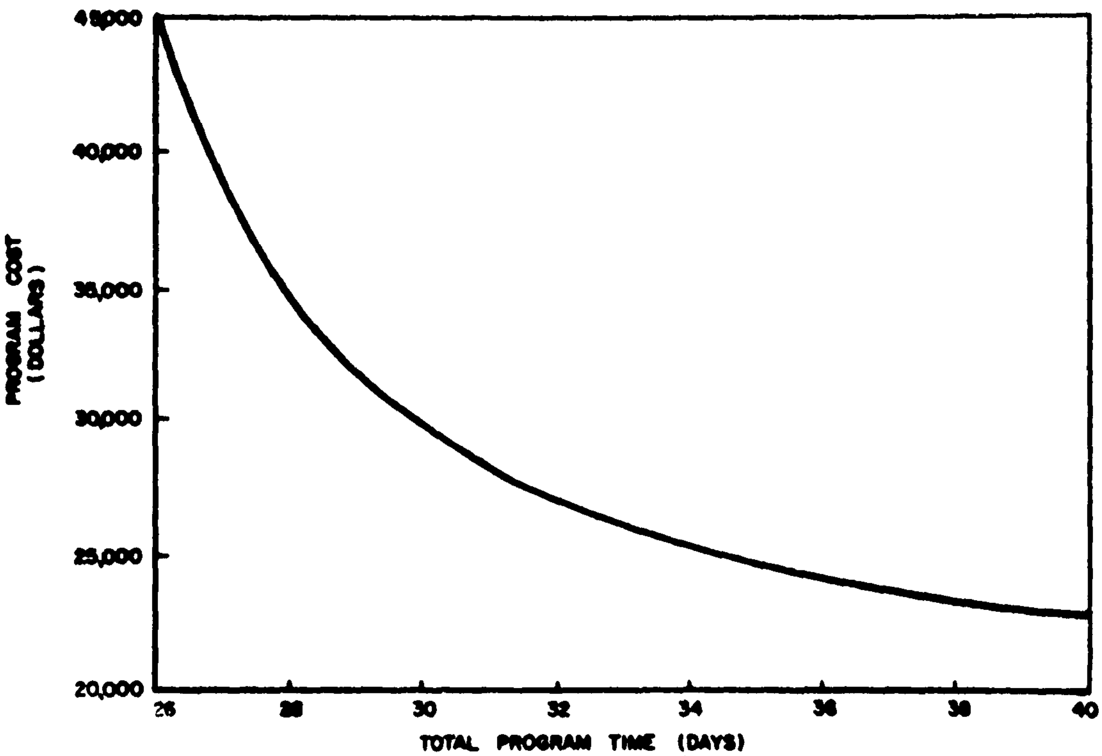
Figure 8 and Table II illustrate a balanced resource allocation such that the sum of time-cost slopes of activities terminating at a junction is equal to the sum of time-cost slopes of activities commencing at the same junction, for all intermediate junctions, with a small rounding error.

3.0 BALANCING A NETWORK UNDER CERTAINTY FOR A VARIABLE PROGRAM TIME

Since it is possible to find a minimum cost solution for any given total program time, T, it is also possible to vary T parametrically and plot the costs for various Ts. Assuming it is possible for the contractor to deliver the product or system represented by the network earlier than scheduled, the T-cost function will reach a minimum, but not turn up, as illustrated in Figure 9.

Figure 9

IA -9816



THE T-COST FUNCTION

This figure represents the constants of Table I, with T, total program time, varied parametrically from 26 to 40 days at two-day intervals.

The figure illustrates that total program cost declines monotonically as a function of increasing T, but at a decreasing rate of decrease.

Such a function might be presented to the decision-maker as an estimate of what he would have to pay to save time on his program. Assuming the decision-maker is able to evaluate the value of earlier delivery of the system or subsystem represented by the network, he is then able to balance the value and cost of early delivery.

At the minimum cost T, some activities should be to the left of their time-cost minima because the slack in a network in fact costs money.* Thus, if the slack is taken at the beginning of a slack activity (postponing the start of the activity as long as possible) there will be a certain amount of inflation in the costs when they are incurred. Alternatively, if the slack is left at the end, there will be inventory holding costs for carrying the completed sub-assemblies in stock. Thus, at the minimum cost T, there will be a balancing of higher processing costs along nonslack activities against lower slack costs in the slack activities; and this balance cannot be at the minima of all activities unless there is no slack in the system at all.

4.0 BALANCING A NETWORK UNDER UNCERTAINTY

This section is limited to an investigation of some of the effects of introducing into a network uncertainty in the time of completion of one activity.

This initial investigation of uncertainty is performed through the use of a serial network. A simple serial network is presented in Figure 10, and its associated constants are presented in Table III.

Figure 10

A BALANCED HYPOTHETICAL SERIAL NETWORK

This network was balanced using the procedure of section 2, and the results are shown in Figure 10 and Table III.

---

*Slack was, however, assumed to be costless in the hypothetical examples.

Table III

	CONSTANTS				SOLUTION		
Activity	a_{ij} (\$'S)	b_{ij}(\$/DAY)	c_{ij}(\$-DAYS)	d_{ij}(DAYS)	t_{ij}(DAYS) Initial	t_{ij}(DAYS) Optimal	Time-Cost Slopes (\$/DAY)
1-2	1,000	100	5,000	4	5	7.17	-397.39
2-3	1,000	120	6,000	5	6	8.41	-397.39
3-4	1,000	80	4,000	7	8	9.89	-397.39
4-5	1,000	50	3,000	6	7	8.59	-397.39
5-6	1,000	150	8,000	3	4	6.82	-397.39
6-7	1,000	110	6,000	6	7	9.44	-397.39
7-8	1,000	120	7,000	6	23	9.68	-397.39

Hypothetical Serial Network Under Conditions of Certainty

Uncertainty is now introduced into one activity at a time. The insertion of uncertainty into activities 2-3, 4-5, and 6-7 is tested alternatively. The relevant probability function is presented in Table IV,

Table IV

t^p_{ij}	Probability
$t^a_{ij} - 2$	.25
t^a_{ij}	.50
$t^a_{ij} + 2$	.25

where t^p_{ij} is the actual, ex post time of activity ij, known only after the completion of the activity, and t^a_{ij} is the ex ante expected time of activity ij.

The cost of the uncertain activity C^p_{ij}, in terms of both t^a_{ij} and t^p_{ij} is now defined as follows:

$$C^p_{ij}\ (t^a_{ij},\ t^p_{ij}) = a_{ij} + b_{ij}\ t^p_{ij} + \left(\frac{c_{ij}}{t^a_{ij}-d_{ij}}\right)\ \frac{t^p_{ij}}{t^a_{ij}};\ d_{ij} < t^a_{ij} \leq \sqrt{\frac{c_{ij}}{b_{ij}}} + d_{ij};$$

$$C^p_{ij}\ (t^a_{ij},\ t^p_{ij}) = a_{ij} + b_{ij}\ t^p_{ij} + \left(\frac{c_{ij}}{\sqrt{\frac{c_{ij}}{b_{ij}}}}\right)\ \frac{t^p_{ij}}{t^a_{ij}};\ t_{ij} > \sqrt{\frac{c_{ij}}{b_{ij}}} + d_{ij}, \tag{3}$$

We suggest, for future research in this area, that a general PERT network, with uncertainty in each activity except dummy activities at the end of terminal activities (introduced to enable the reduction of the total program time to T), be tested through a Monte Carlo model. A negative discount rate could be introduced to move junction times toward the beginning of the network, and alternative discount rates could be tested to see which provides the best cost *ex post*. The network would be solved in each iteration under the techniques of section 2, except for the use of the discount rate. A positive discount rate could be used in appraising the results to represent the fact that costs in the real world should be discounted. It is interesting to note that the avoidance of a discount rate in the case of certainty would tend to compensate for the bias which is introduced by failing to recognize the fact that the t_{ij}s are in fact uncertain.

5.0 SUMMARY

Under the assumption of continuous upward-concave time-cost functions for activities and under the assumption of a given total program time, an iterative algorithm was developed to determine the optimal allocation of time in a PERT network. In this algorithm, junctions are balanced one at a time by equating the sum of time-cost slopes of activities terminating in the junction with the sum of time-cost slopes of activities commencing in the junction, in absolute value. Junctions are balanced in turn, with other junctions held constant at their last computed values, and the network is solved iteratively until a solution is obtained in which all junctions are simultaneously in balance.

A parametric minimum cost solution for alternative total program times was also derived. This solution presents to the decision-maker the incremental cost of program acceleration.

Finally, uncertainty in the time of completion of an activity was investigated in a simplified network. The conclusion was reached that the introduction of uncertainty causes a shift of junction times in the network toward the beginning of the program, as compared with the case of certainty.

E. B. Berman

E. B. Berman

EBB/lc

Attachments: Bibliography, Distribution List

Figures:	1A-9515	- page 1	1A-9519	- page 6
	1A-9524	- page 2	1A-9522	- page 8
	1A-9517	- page 3	1A-9521	- page 9
	1A-9523	- page 4	1A-9516	- page 10
	1A-9518	- page 5	1A-9520	- page 11

where C^p_{ij} is the ex post cost of uncertain activity ij. These cost functions assume that the allocation of time to various activities in the network is reviewed only after the completion of the uncertain activity, not during the activity; so that once an allocation of resources to the uncertain activity is determined, the effect of a longer or shorter t^p_{ij} is only to increase or decrease the linear component of cost and to extend or shorten the use of allocated resources, $\frac{c_{ij}}{t^a_{ij}-d_{ij}}$. Since the function $C^p_{ij}\ (t^a_{ij},\ t^p_{ij})$ is linear in t^p_{ij}, given t^a_{ij}, it is possible to write:

$$C^p_{ij}\ (t^a_{ij},\ t^p_{ij}) = C_{ij}\ (t^a_{ij}) \tag{4}$$

or in other words, the expected cost of the uncertain activity is equal to the cost under the case of certainty using t^a_{ij} in place of t_{ij}.

It is assumed that the remaining portions of the network are reprogrammed after the completion of the uncertain activity and are balanced under the constraint of the remaining total program time available.

The optimal ex ante network allocations under this simple case of uncertainty are shown in Table V. The solution under certainty is repeated for comparison.

Table V

Activity	Case of Certainty		Case of Uncertainty					
			2-3 Uncertain		4-5 Uncertain		6-7 Uncertain	
	t_{ij}	Slopes	t_{ij}	Slopes	t_{ij}	Slopes	t_{ij}	Slopes
1-2	7.171	-397.4	7.155	-402.2	7.134	-409.2	7.063	-432.9
2-3	8.405	-397.4	8.390	-402.2	8.367	409.2	8.294	-432.9
3-4	9.895	-397.4	9.900	-395.5	9.860	-409.2	9.793	-432.9
4-5	8.590	-397.4	8.595	-395.5	8.556	-409.2	8.492	-432.9
5-6	6.823	-397.4	6.830	-395.5	6.871	-383.9	6.705	-432.9
6-7	9.439	-397.4	9.445	-395.5	9.485	-383.9	9.324	-432.9
7-8	9.678	-397.4	9.685	-395.5	9.727	-383.9	10.328	-253.7
COST:	$24,790		$24,851		$24,887		$25,071	

Balanced Serial Network Under Uncertainty

In analyzing the results presented in Table V, the reader should note that wherever the uncertainty is located, the effect is always to shift intermediate junctions toward the beginning of the program, relative to the case of certainty. This shift raises (in absolute value) time-cost slopes preceding the end of the uncertain activity, and lowers time-cost slopes (in absolute value) following the end of the uncertain activity, reflecting a shift of all junctions towards the beginning of the program. This effect is a form of insurance, protecting against a very steep climb up the time-cost functions of activities following the end of the uncertain activity. Therefore, as a general conclusion, we may note that there is some evidence that uncertainty tends to shift the time of junctions in a PERT network toward the beginning of the program.

BIBLIOGRAPHY

1. J. E. Kelley, Jr., M. R. Walker, "Critical Path Planning and Scheduling," Proc. EJCC Conf., 1959, 160-173.

2. D. R. Fulkerson, "A Network Flow Computation for Project Cost Curves," Mgt. Sci., 7 (1961), 167-178.

3. J. E. Kelley, Jr., "Critical Path Planning & Scheduling: Mathematical Basis," Oper. Res. 9 (1961), 296-320.

4. A. Vazsony, "An On-Line Management System Using English Language," presented at the Western Joint Computer Conference, L.A., 9-11 May 1961.

5. M. C. Salveson, "Principles of Dynamic Weapon Systems Programming," Naval Research Logistics Quarterly, Vol. 8, No. 1, March 1961.

6. L. Alpert, D. S. Orkand, "A Time Resource Tradeoff Model for Aiding Management Decisions," ORI, Inc., TP-12, May, 1962.

7. R. E. Beckwith, "A Cost Control Extension of The PERT System," Aeroneutronic Division, Ford Motor Company, October, 1961.

8. D. R. Morrison, "Cost Functions for Systems with Spare Components," Oper. Res., 9 (1961), 688-694.

DISTRIBUTION LIST

MITRE

Document Control Center (10 copies)

C. W. Halligan
O. O. Sulkala - Record
F, G, H, I

D-2

S

D-18

All Staff
N. Waks (2 copies)
D. F. Votaw, Jr. (2 copies)
E. B. Berman (65 copies)

D-19

E. Neidorf (4 copies)

D-27

J. D. Porter (4 copies)
L. Suyemoto

ESD

ESOPT
Maj. Robert S. Gaines, ESOPS-2
DCS/Plans and Operations, ESO
Director of Programs, ESOP
Technical Information Services Division (ESAT)
Deputy for Systems Management, ESS (3 copies)
Deputy for Advanced Planning, ESL

AFSC

Directorate of Cost Analysis, SCCC (3 copies)
Col. J. J. Bennett, SCCS (4 copies)